Introduction to Python on ASU RC HPC Systems

Jason Yalim

2020-02-03

Why
High Performance Computing (HPC)

Pool of capital investment into state of the art hardware and software

Team(s) of system administrators and IT to maintain hardware and software

Advantages of remote computation

Data Plague

Suppose a genomics researcher is sequencing small datasets, 1000 each 500 MB

But new datasets are in, 20x as large (10 GB), 2x as many

500 MB file may already be a burden on most machines!

https://epcced.github.io/hpc-intro/010-hpc-concepts/

The Statistics Student

Suppose statistics student requires cross validation study

1000 runs required

Each run takes 1 hour on laptop

41 days and 16 hours of compute time!

https://epcced.github.io/hpc-intro/010-hpc-concepts/

Idle Resources

Suppose a student with 4 core laptop is running a 2D PDE solver in serial

Needs to go to 3D and parallel solver can be implemented

Convert laptop into quasi-permanent solver/heater?

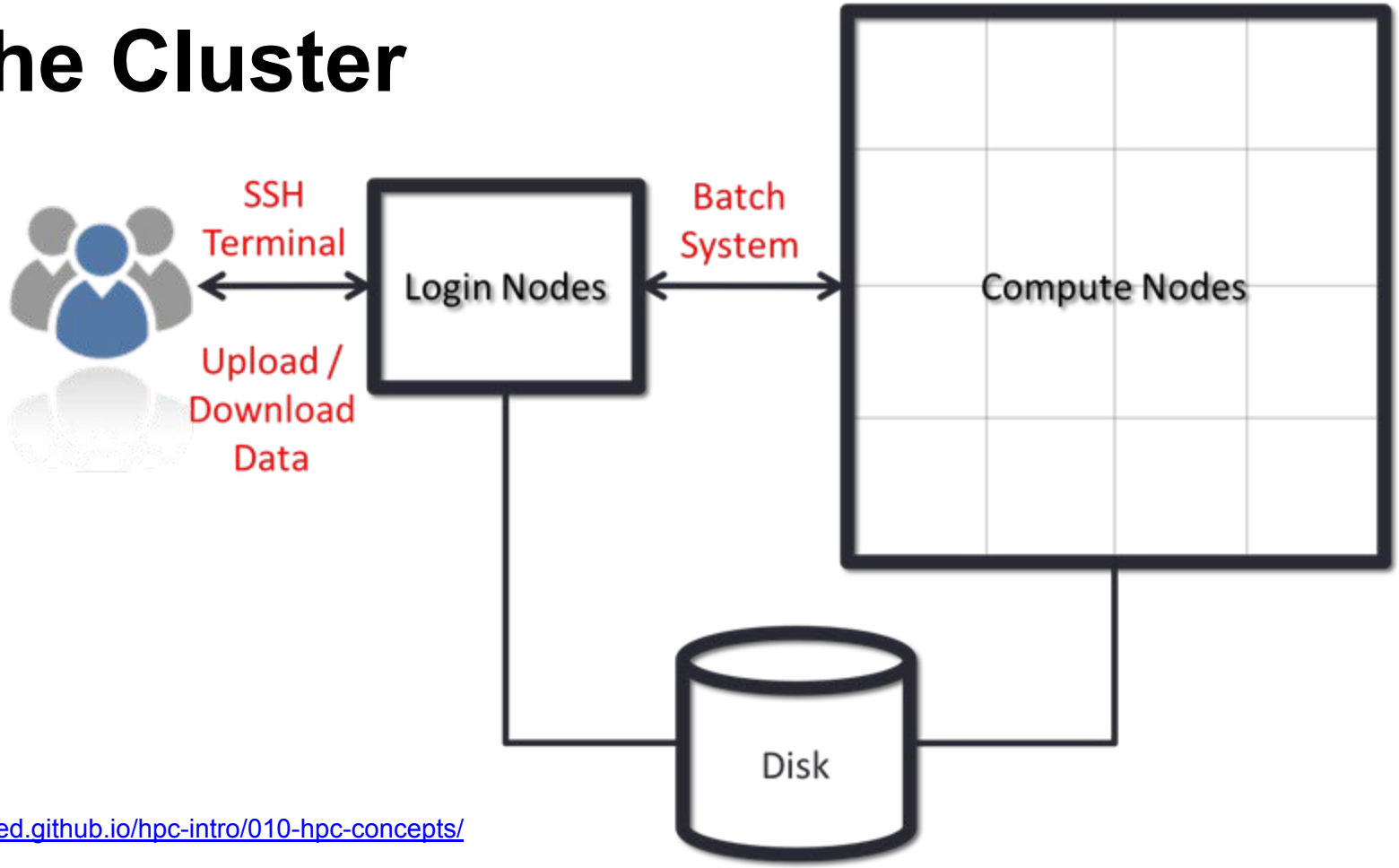
https://epcced.github.io/hpc-intro/010-hpc-concepts/

Considerations for HPC

- **Speed.** With many more processing cores, often with higher performance specs, than a typical laptop or desktop, HPC systems can offer significant speed up.
- **Volume.** Many HPC systems have both the processing memory (RAM) and disk storage to handle very large amounts of data.
- **Efficiency.** Many HPC systems operate a pool of resources that are drawn on by a many users. In most cases when the pool is large and diverse enough the resources on the system are used almost constantly.
- **Cost.** Bulk purchasing and government funding mean that the cost to the research community for using these systems in significantly less that it would be otherwise.
- **Convenience.** There's no need to tie up your own computer for hours when you can use someone else's instead.

https://epcced.github.io/hpc-intro/010-hpc-concepts/

The Cluster

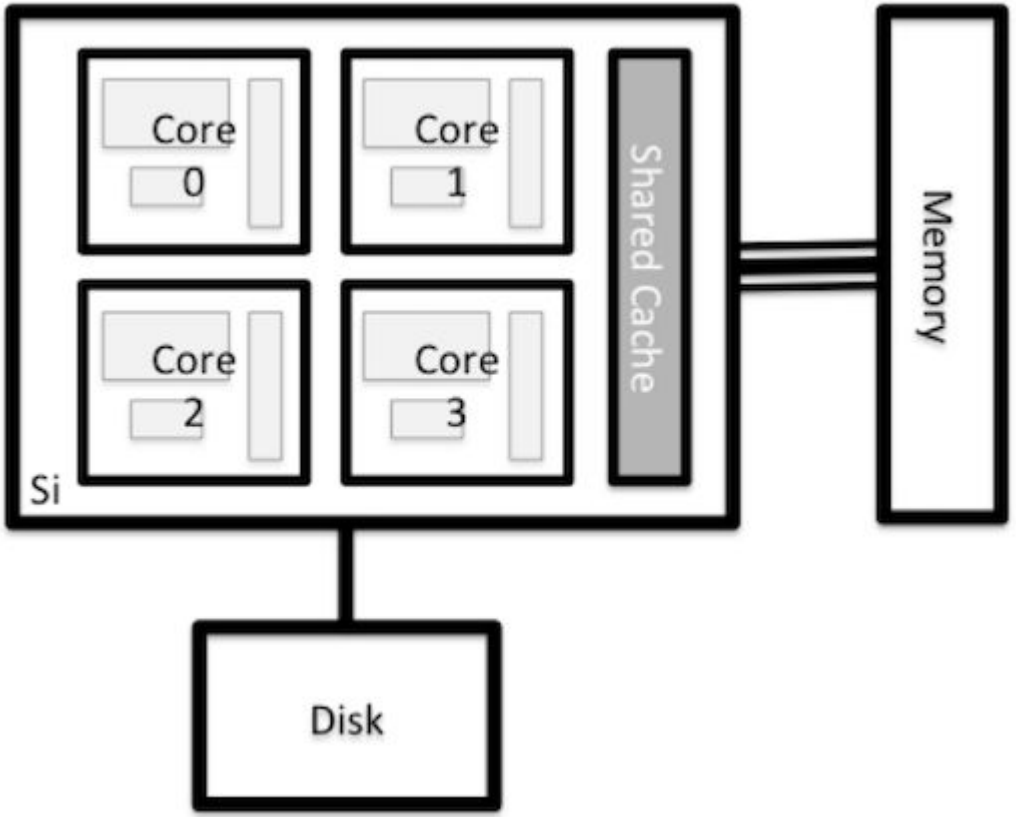


https://epcced.github.io/hpc-intro/010-hpc-concepts/

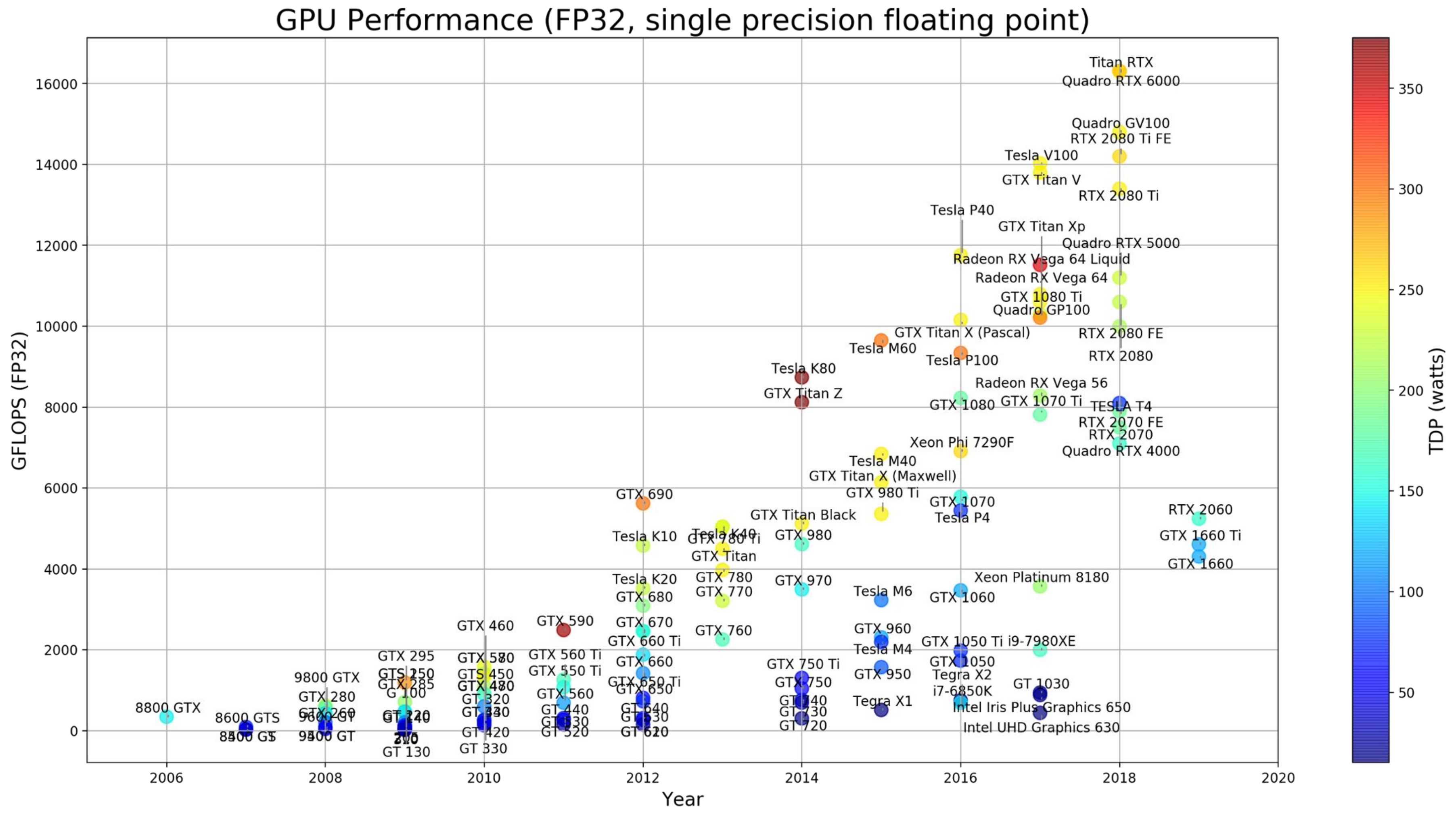
Compute Nodes

Agave has nearly 350 compute nodes

Includes nearly 200 GPUs



https://epcced.github.io/hpc-intro/010-hpc-concepts/



GPU Performance (FP32, single precision floating point)
GFLOPS (FP32)
Year
TDP (watts)
0
2000
4000
6000
8000
10000
12000
14000
16000
2006
2008
2010
2012
2014
2016
2018
2020
50
100
150
200
250
300
350
Titan RTX
Quadro RTX 6000
Quadro GV100
RTX 2080 Ti FE
Tesla V100
GTX Titan V
RTX 2080 Ti
Tesla P40
GTX Titan Xp
Quadro RTX 5000
Radeon RX Vega 64 Liquid
Radeon RX Vega 64
GTX 1080 Ti
Quadro GP100
GTX Titan X (Pascal)
Tesla M60
Tesla P100
RTX 2080 FE
RTX 2080
Tesla K80
GTX Titan Z
Radeon RX Vega 56
GTX 1080
GTX 1070 Ti
RTX 2070 FE
RTX 2070
Xeon Phi 7290F
Quadro RTX 4000
Tesla M40
GTX Titan X (Maxwell)
GTX 980 Ti
GTX 690
GTX 1070
Tesla P4
GTX Titan Black
RTX 2060
Tesla K10
GTX 980
GTX 1660 Ti
GTX Titan
GTX 1660
Tesla K20
GTX 780
GTX 970
Xeon Platinum 8180
GTX 680
GTX 770
Tesla M6
GTX 1060
GTX 460
GTX 590
GTX 670
GTX 760
GTX 960
GTX 660 Ti
GTX 1050 Ti
i9-7980XE
Tesla M4
GTX 295
GTX 560 Ti
GTX 660
GTX 750 Ti
GTX 1050
GTX 550 Ti
GTX 750
GTX 950
Tegra X2
9800 GTX
GT 1030
i7-6850K
GTX 280
G 100
GTX 560
Tegra X1
8800 GTX
8600 GTS
GT 440
Intel Iris Plus Graphics 650
GT 730
GT 720
Intel UHD Graphics 630
GT 420
GT 520
GT 130
GT 330

Important Links

links.asu.edu/getHPC <- create an account

links.asu.edu/docs <- our documentation

links.asu.edu/MailingList <- our mailing list

links.asu.edu/RCpy <- link to workshop github

rcstatus.asu.edu <- system status pages

login.rc.asu.edu <- new webapp interface to Agave

Upcoming Workshops

Python, Data and HPC (links.asu.edu/pyHPC2)

- Feb 10, 3-4 PM Biodesign A L10-14

Python, ML and HPC (links.asu.edu/pyHPC3)

- Feb 24, 3-4 PM Biodesign A L10-14

Python, DL and HPC (links.asu.edu/pyHPC4)

- Mar 9, 3-4 PM Biodesign B Auditorium

Conclusion

For any assistance please contact: support@hpchelp.asu.edu

Office Hours: **GWC546 1-4 PM Tuesdays and Wednesdays**

login.rc.asu.edu

Supplemental Slides Follow

The Scheduler

1. Transfer input datasets to the HPC system (via the login nodes)

2. Create a job submission script to perform your computation (on the login nodes)

3. Submit your job submission script to the scheduler (on the login nodes)

4. Scheduler runs your computation (on the compute nodes)

5. Analyse results from your computation (on the login or compute nodes, or transfer data for analysis elsewhere)

https://epcced.github.io/hpc-intro/010-hpc-concepts/

- **High Performance Computing (HPC).** Computing resources that allow people to solve their problems faster or treat larger problems than they would be able to using standard computing resources (e.g. a laptop, desktop or workstation). Usually implies some sort of *parallel computing*.
- **Parallel Computing.** The use of computing resources in parallel to speed up computation or treat larger computational problems.
- **Supercomputer.** Typically used to describe a very large HPC resource such as those found on the Top500 list. Often uses the same technology as *compute clusters* but at a larger scale.
- **Compute Cluster.** Typically used to describe a smaller HPC resource than those referred to as *supercomputers*. Usually use exactly the same technology as supercomputers but on a smaller scale.
- **High Throughput Computing (HTC).** A subset of parallel computing where computing resources are used in parallel on many independent sub-tasks to increase the rate at which computation can be performed. For example, varying an input parameter (or input data) to a computation and running many copies simultaneously.
- **Cloud Computing.** Using remote computing resources *on demand*. Often associated with using public cloud computing resources provided by large internet corporations.